http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 12 (19): 5367-5371, 2013
ISSN 1812-5638 / DOI: 10.3923/itj.2013.5367.5371

An Exploration of Recommender Systems Based on Url Hierarchy Relationships

[1]Zhou Meilin, [1,2]Xu Yan, [1]Han Siyao and [1]Zhao Yaqing
[1]School of Information Science, Beijing Language and Culture University, Beijing, 100083, People's Republic of China
[2] Institute of Computing Technology, Chinese Academy of Sciences, 100190, Beijing, People's Republic of China

Abstract: Recommendation systems have become an important research area since the appearance of the first studys on collaborative filtering in the mid-1990 (Hill *et al*., 1995). Several existing methods came from the perspective of content and each of them has advantages and disadvantages. In this study, we proposed a recommendation method based on url-based hierarchical relationships. We calculate level of candidate pages after the analysis of the hierarchical relationship between pages. Different from traditional methods, this can implicitly use pages category information. At the same time considering the hierarchical relationship of web page to get more information. Thus, to a certain extent, resolve data sparsity problem. Experiment shows that, in this way, result is better than those hierarchies are not considered in recommended precision.

Key word: Recommendation algorithm, URL hierarchy, web mining

INTRODUCTION

Web is a set that full of noisy data and in tissue disorders. On the one hand there exist numerous documents, graphs, images and so on, which showed wed data diversity. On the other hand the users, having different backgrounds, interests and purposes, are free to link to any of these sites. Web users are thus exhibiting diversity characteristics. Internet has brought great convenience and rich information resource to users and at the same time produced the following problems to be solved.

Nowadays there are various recommended methods with the following three main, Content-based approach (Mobasher, *et al*., 2000), based collaborative approach (Sarwar *et al*., 2001), based on association rules. All these methods are based on user clicks or web content without considering another important factor, the structure of web information. So, we propose the use of hierarchy to express the relationship of the pages. Advantages are as follows:

- Express the type information of the page implicitly. By using this hierarchical relationship, we can define the type of a web page and express the connection with other web pages without classification or clustering
- Father-child node referred web pages, regarded as ordinary pages in those methods previously used, can't be treated equally. In fact for a leaf node, its parent node may be its class information or navigate to contents page to a certain degree. Thus node weights should be taken into consideration

Hierarchy web can provide more information. This to a certain extent solved data sparsity problem of which collaborative filtering faced with.

RELATED WORK

By implementing several commonly used recommended method mentioned before, such as association rules, clustering and so on. Here we analyzed these methods in detail.

Sarwar and other researchers (Sarwar *et al*., 2000) put forward the recommendation system based on association rule mining. Association rule method develop production rules according to user's static characteristics and dynamic properties, calculate the web pages the users may be interested in but haven't visit through rule matching, then order pages in accordance with the rules of support degree. At last recommend the users the top N pages (Wang *et al*., 2005). Apriori algorithm is the most commonly used algorithm. The basic idea of this method is to generate frequent itemsets which is used to produce rules under a certain confidence level and the recommended process is dependent on the rules. For example, A, B and C are three web pages, we get the

Corresponding Author: Zhou Meilin, School of Information Science, Beijing Language and Culture University, Beijing, China

frequent itemsets {A, B, C} under a certain support, then generate rules A, B => C. Though we can find the frequent itemsets accurately and then create rules in this method, it also has some defects:

- Recommended page must exist in rules, that is to say the page must have ever been visited. So, association rules have no preference for the new page
- Rules are from frequent item sets. Frequent item sets, by literally can see is the frequent access page collections visited at the same time. But for some pages, such as news web pages, are accessed only once. Under the same condition, it seems unfair to compare with other categories frequently visited. So, the definition of frequency is not representative

Collaborative filtering recommendation technology is aimed at mining the relationship between user information and project information, then produce recommendation results according to their similarity. Karypis and others provide recommendation service on the basis of their similarities, thereby improve the quality of recommendation effectively. Aggarwal *et al.* (1999) proposed an optimization algorithm through graph search of user nearest neighbors. Yu *et al.* (2003) used different weights to nearest neighbors and items to improve the quality of recommendation. Collaborative filtering is a more successful method in current personalized recommendation system, but Data Sparsity is the main reason that led to the decrease of the quality of the recommendation system (Wang *et al.*, 2005). In the document (Breese *et al.*, 1998), Collaborative recommendation algorithm can be divided into two categories: Based on the heuristic algorithm (Adomavicius and Tuzhilin, 2005) and based on the algorithm of the model (Adomavicius and Tuzhilin, 2005). They also have some shortcomings:

- Heuristic method can't avoid data sparseness matrix problem when faced with large-scale users and page data. Most study of the collaborative filtering are mostly based on the user-filling project matrix and thus reduce the sparse matrix, getting much more accurate calculation results of nearest neighbor
- Take clustering as a model of model based algorithm. Firstly, gather N point to K class first. Recommendation is based on the relationship in the class, thinking the inside of the same web page class have more similar representative. So if the user read a web page in the class, program is likely to recommend other pages ignoring another important factor, that is the distance between the two points is not necessarily very near. Three classes shown in the Fig. 1, the distance between the class | AB | is clearly less than the Within the class distance |BC|

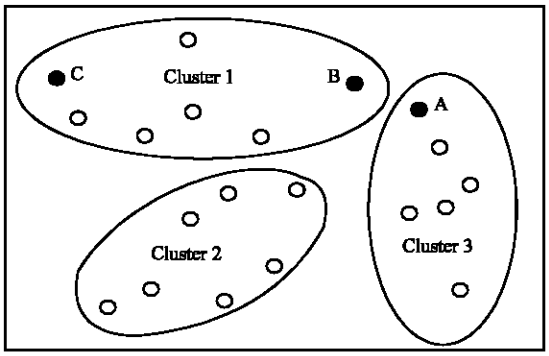

Fig. 1: Clustering distribution

Another recommendation technology is based on content filtering to recommend information by comparing the resources and the user interest files. The key problem is to extract the user interest and similarity calculation. It can accurately locate specific content of the Web with some limitations, since analyzing the content information of resources is a must, such as music, images, video and other information. Thus, it's difficult to analyze the quality of information and provide diversified recommendation.

ALGORITHM DESIGN

Previous data matrixes are usually obtained from simple user behaviors, such as user clicks on one web page, user browsing time on one web page. However, this does not fully represent their similarity, because there is no direct relation between page similarity and clicks. For example, a newly released movie trailers and news of a plot of public concern, their clicks and browsing time are probably very high, then the similarity calculated from matrixes may also be great, but there is no direct relationship between these two pages. We propose a recommend method based on URL hierarchy relationships. With web hierarchy information, child nodes of the same parent would have a certain correlation and then a relatively high degree of similarity can represent the user's interest.

Hierarchical structure: As we know, on the server side, web storage is carried out in accordance with the folder directory, for example, the following table is a URL directory. We can clearly see the directory structure on the server side of their information from the URL. The folder structure is a hierarchical structure.

Table 1: URLs of http://www.db-net.aueb.gr

URL	Description
/courses/datamining/index.html	Data mining course
/courses/filesdb/btree.htm	Tutorial on B-trees
/courses/filesdb/b-trees2002.htm	Coursework on B-trees
/michalis/phds_new.html	Announced PhD positions
/michalis/publications.html	List of publications of M.vazirgiannis
/people/michalis.html	Home page of professor M. vazirgiannis

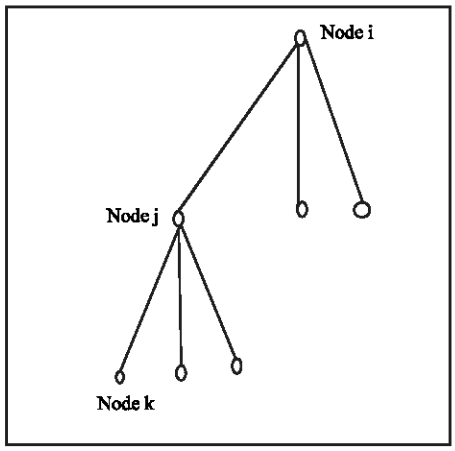

Fig. 2: Schematic page hierarchy

To some extent, the hierarchical structure indicates some category information of these pages. Pages of the same layer and pages of adjacent layers both have certain relevance. This is the premise that we put forward the following idea. Here we consider only adjacent pages and transfer up to two times. A simple schematic section page hierarchy is shown in Fig. 2. Node j is to be calculated, that is, the recommendation candidate page. Node k is the child node of node j; node l is the parent node of node j (as shown in Fig. 2).

Algorithm description: In order to use the hierarchical relationship, first of all, we use hierarchy information to build user-item matrix, then calculate all scores of the pages to be recommended, finally sort them to obtain the recommended result.

Definition 1: Visit coverage:

$$\text{Cover}(i,j)=\frac{|\text{vst}(\text{cld}(j))|}{|\text{cld}(j)|}$$

The above equation describes the visit coverage of a node j is the number of its accessed child nodes (i) over the total number of its child nodes.

Definition 2: Weight matrix M. To calculate the similarity between pages, we need the user-item matrix to obtain:

$$M=\underbrace{\begin{bmatrix} w_{11} & w_{21} & \cdots & w_{m1} \\ w_{12} & w_{22} & \cdots & w_{m2} \\ \cdots & \cdots & w_{ij} & \cdots \\ w_{1n} & w_{2n} & \cdots & w_{mn} \end{bmatrix}}_{\text{item}}\Big\}\text{user}$$

- In the past, we simply consider the user i clicks on page j; we put this value as our baseline. Where, w_{ij} = click (i, j)
- Here we define a new page weight matrix element:

$$w_{ij}=\text{click}(i,j)+\frac{\Sigma_{k\in cld(j)}\,\text{cllick}(i,k)}{|\text{vst}(\text{cld}(j))|}\times\text{cover}(i,j)+\frac{\Sigma_{l\in parent(j)}\,\text{cllick}(i,l)}{|\text{cld}(l)|}$$

So, that we get the user-project matrix mentioned above, that can link a node j with his parent node and child node.

Detailed explanation: The formula consists of three parts, namely, the click times of the node itself, the transfer clicks of its child nodes and the transfer clicks of its parent node.

In the first part, the click times are always taken into account by the common methods, combined with hierarchical tree information, it can be used as base values □□ of our weights.

The second part is the transfer of child nodes. Visiting a page, for example, a technology news page, is actually a reflection of the extent of users' interest. If the user frequently visits such technology news, the recommended weight of this kind of news page should be increased. Specifically, in the diagram above, node k stands for a technology news web page, node j is the technology news directory, then node k should transfer to node j. Because if the recommended results contain node j, users can visually see all news titles to quickly find other news they want to see. The equation is:

$$\frac{\Sigma_{k\in cld(j)}\,\text{cllick}(i,k)}{|\text{vst}(\text{cld}(j))|}\times\text{cover}(i,j)$$

The first half of the equation represents the average clicks of those visited child nodes of node j. the second half part, which has been explained in definition 1, is the coverage. That says the greater the coverage is, these child nodes are more likely accessed and the parent nodes are more convincing.

The third part is the transfer of parent node. From earlier example, we already know that node j is a technology news directory and its parent node l may be a directory of all news items, including science news, financial news, entertainment news, etc. We believe that

after a user visit the parent node l, he has equal possibilities accessing to other links in the directory, which says he would probably visit all child nodes j. This is the visit transitive of the parent node. The equation is:

$$\frac{\Sigma_{l\in parent(j)} cllick(i, l)}{|cld(l)|}$$

It represents the click weight of each child node that transferred from their parent nodes l, (here node l are the parent nodes of node j).

Definition 3: In order to get all the pages to be recommended for a user, we need to calculate the scores of the pages to be recommended and then sort them to get the top n pages to recommend:

$$score(U_i, p_j) = \sum_{\substack{p_t \in U_i \\ p_t \notin U_i}} sim(p_i, p_j)$$

where, U_i is the page set that have been visited by user i. Therefore, the recommended score of page p_j is converted to the similarity between page p_j and all pages user i has visited. At this point, we should consider the relationship between page p_i and page p_j. As shown in Fig. 3.

From the above chart we can see all possible combinations of the relationships between nodes p_i and p_j. We have already mentioned two preconditions: Only consider the transfer relationship between adjacent nodes and transfer up to two times. Simple relationships between p_i and p_j include parallel nodes, parent-child and child-parent relationships. Slightly more complicated relationships between p_i and p_k are parallel nodes, the nodes to be recommended are p_k's parent nodes or child nodes. Other cases, such as two-story parent nodes, two-story child nodes, parallel but not has the same parent nodes, because of the far distance and un-obvious transitive, we would like to take them into other situation To summarize, five different situations should be considered in the calculation of the similarity based on the hierarchy:

$$sim(p_i, p_j) = \begin{cases} cos(p_i, p_j) + \frac{1}{|cld(p_1)|-1}, (p_i, p_j \in cld(p_1)) \\ cos(p_i, p_j) + \frac{1}{cld(p_i)}, (p_j \in cld(p_i)) \\ cos(p_i, p_j) + \frac{1}{cld(p_j)}, (p_i \in cld(p_j)) \\ cos(p_i, p_j) + cos(p_i, p_k) * \frac{1}{cld(p_k)}, (p_i \in cld(p_1), p_j \in cld(p_k)) \\ cos(p_i, p_j), (other\ \ Situations) \end{cases}$$

The above equation provides several similarity calculation solutions respectively, in accordance with hierarchical relationship in the figure.

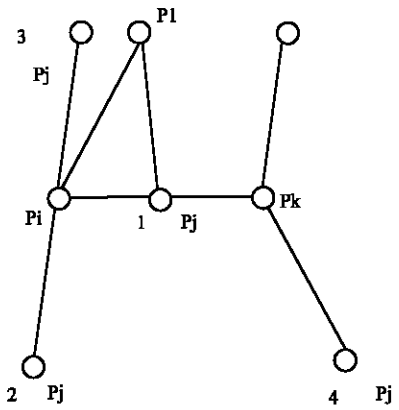

Fig. 3: Hierarchical relationship of p_i and p_j

Table 2: Recommended accuracy rate changes with N

No.	Baseline	Proposed method	Improvement
5	0.817664	0.849003	0.031339
6	0.826211	0.85755	0.031339
7	0.846154	0.868946	0.022792
8	0.851852	0.868946	0.017094

Experiments: Data used in this study is from the access log of Beijing Language and Culture University web site from 0:00 on June 1, 2011-24:00, the log size is about 187M. After log cleaning, user identification, session identification, such these pre-processing ,we get the number of users, number of pages and the number of sessions, that is 7333, 3731 and 8303 respectively and then transformed the access log into user session log representation according to experimental needs.

To investigate the impact of the recommended network hierarchy effect, we have designed a set of comparative experiments. We implemented hierarchical methods and non-hierarchical methods on the same recommend method simultaneously and compared the accuracy of the results. We obtained the first N pages using the TOP N highest-scoring method.

From the graph we can see that the hierarchical method got a higher accuracy in the results than the one without hierarchical information. Meanwhile, it is clear that with the increasing value of N, the recommended pages were increasing and the accuracies of both methods were gradually rising. When N increases, the degree of accuracy increases flatten.

Thus, we conclude that the use of hierarchical relationships in web recommendation system can improve the quality of recommendation. Increasing the recommend number in a certain degree and using large databases will play a positive role.

We also verified two other advantages of the web hierarchy by experiments: weakening data sparseness problem, which has plagued the collaborative filtering to a certain extent; recommend pages can be un-visited pages.

CONCLUSION

Existing recommend methods include content-based methods (Mobasher *et al*., 2000), collaborative filter methods (Sarwar *et al*., 2001), association rules methods and so on. They are only based on the user clicks or the page content information, without considering the hierarchical structure of the pages. This study proposes a recommend method based on web URL hierarchical structure to analyze hierarchical relationships between pages, calculate the recommend degrees of pages to be recommended with different structures and then do the final recommendation. The difference of this method from the traditional method is that it implicitly expressed the category information between pages, differentiated the weights of different level pages. Taking into account the hierarchical relationship of the pages can get more information and weaken the data sparseness problem to some extent.

ACKNOWLEDGMENTS

This study is supported by National Key Technology Research and Development Program of China (2012BAH39B02), National Natural Science Foundation of Beijing. No.: 4122076 and Fundamental Research Funds for the Central Universities(13YCX176). We would like to thank the reviewers for providing valuable comments and advices.

REFERENCES

Adomavicius, G. and A. Tuzhilin, 2005. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Trans. Knowledge Data Eng., 17: 734-749.

Aggarwal, C.C., J.L. Wolf, K.L. Wu and P.S. Yu, 1999. Horting hatches an egg: A new graph-theoretic approach to collaborative filtering. Proceedings of the 5th International Conference on Knowledge Discovery and Data Mining, August 15-18, 1999, San Diego, CA., USA., pp: 201-212.

Breese, J.S., D. Heckerman and C. Kadie, 1998. Empirical analysis of predictive algorithms for collaborative filtering. Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence, Jul 24-26, 1998, Madison, WI., pp: 43-52.

Hill, W., L. Stead, M. Rosenstein and G. Furnas, 1995. Recommending and evaluating choices in a virtual community of use. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems Denver, May 7-11, ACM Press/Addison-Wesley Publishing Co., Colorado, United States, pp: 194-201.

Mobasher, B., R. Cooley and J. Srivastava, 2000. Automatic personalization based on web usage mining. Commun. ACM., 43: 142-151.

Sarwar, B., G. Karypis, J. Konstan and J. Reidl, 2001. Item-based collaborative filtering recommendation algorithms. Proceedings of the 10th International Conference on World Wide Web, May 1-5, 2001, Hong Kong, China, pp: 285-295.

Sarwar, B., G. Karypis, J. Konstan and J. Riedl, 2000. Analysis of recommendation algorithms for e-commerce. Proceedings of the 2nd ACM Conference on Electronic Commerce, October 17-20, 2000, ACM Press, Minneapolis, MN, USA., pp: 158-167.

Wang, X., Y. Ling and Y.L. Fei, 2005. Personalization recommendation system based on web log and cache data. Mining, 24: 324-328.

Yu, K., X. Xu, M. Ester and H.P. Kriegel, 2003. Feature weighting and instance selection for collaborative filtering: An information-Theoretic approach. Knowledge Inform. Syst., 5: 201-224.